

LỜI GIỚI THIỆU

Hiện nay, khoa học công nghệ đã và đang chiếm nhiều ưu thế vượt trội. Nó là một cuộc cách mạng lớn góp phần thay đổi toàn xã hội. Để góp phần làm phong phú thêm nguồn tư liệu phục vụ nghiên cứu, học tập cho bạn đọc trong và ngoài ngành Công nghệ thông tin và truyền thông, Nhà Xuất bản Đại học Cần Thơ xin được phép ấn hành và giới thiệu cùng bạn đọc giáo trình “Nhập môn công nghệ phần mềm” do TS. Huỳnh Xuân Hiệp biên soạn. Giáo trình bao gồm 05 chương với 132 trang; Nội dung các chương giới thiệu khái quát công nghệ phần mềm, tiến trình phần mềm, chu trình sống của phần mềm, cách ước lượng chi phí để tính giá trị của các phần mềm. Thêm vào đó, cuối mỗi chương còn có rất nhiều tài liệu tham khảo hữu ích cho bạn đọc. Giáo trình là tài liệu tham khảo có giá trị cho sinh viên các ngành Công nghệ thông tin và bạn đọc muốn tham khảo những vấn đề liên quan đến phần mềm.

Nhà Xuất bản Đại học Cần Thơ chân thành cảm ơn các Tác giả và sự đóng góp ý kiến của quý Thầy Cô trong Hội đồng thẩm định trường Đại học Cần Thơ để giáo trình “Nhập môn công nghệ phần mềm” được ra mắt bạn đọc.

Nhà Xuất bản Đại học Cần Thơ trân trọng giới thiệu đến giảng viên, sinh viên và bạn đọc giáo trình này.

Chân thành cảm ơn!

NHÀ XUẤT BẢN ĐẠI HỌC CẦN THƠ

LỜI NÓI ĐẦU

Quyển giáo trình *Nhập môn Công nghệ phần mềm* mang đến cái nhìn tổng quan về quá trình thực hiện phần mềm cũng như những giai đoạn đặc trưng của quá trình này. Trong nội dung giáo trình cũng đề cập đến một quy trình công nghệ phần mềm tiêu biểu là quy trình RUP (Rational Unified Process). Các kỹ thuật tính toán ước lượng về kích thước phần mềm, chi phí, nhân lực và thời gian thực hiện được đề cập đến một cách khá chi tiết. Kỹ thuật ước lượng trị giá phần mềm theo công văn 3364/UĐCNTT-BTTTT của Bộ Thông tin & Truyền thông cũng được đưa vào giảng dạy như là một nội dung thực tiễn mà sinh viên chuyên ngành máy tính/công nghệ thông tin cần nắm bắt được khi ra trường.

Việc biên soạn giáo trình nhằm mục tiêu giảng dạy cho các sinh viên trình độ kỹ sư máy tính/công nghệ thông tin (các chuyên ngành kỹ thuật phần mềm, hệ thống thông tin,...) trên cơ sở *Bài giảng Nhập môn Công nghệ phần mềm* được tác giả giảng dạy tại Trường Đại học Cần Thơ từ nhiều năm qua và tham khảo các kiến thức, các nguồn tài liệu được công nhận và ứng dụng rộng rãi của các nhà chuyên môn, các chuyên gia và các đồng nghiệp.

Những đóng góp chuyên môn sâu của các đồng nghiệp thuộc Bộ môn Công nghệ phần mềm, Khoa Công nghệ Thông tin & Truyền thông và Hội đồng thẩm định giáo trình của Trường Đại học Cần Thơ đã giúp giáo trình được hoàn thiện hơn rất nhiều.

Trong quá trình biên soạn giáo trình chắc chắn không tránh khỏi có những thiếu sót. Rất mong nhận được sự góp ý, phản hồi từ các bạn sinh viên, các đồng nghiệp và các chuyên gia trong lĩnh vực công nghệ phần mềm để quyển giáo trình này có chất lượng ngày càng tốt hơn.

Cần Thơ, ngày 22 tháng 06 năm 2011

HUỶNH XUÂN HIỆP
PHAN PHƯƠNG LAN

MỤC LỤC

CHƯƠNG 1. TỔNG QUAN	1
1. PHẠM VI	1
2. TIẾN TRÌNH PHẦN MỀM	3
3. CÁC MÔ HÌNH CHU TRÌNH SỐNG CỦA PHẦN MỀM	6
3.1 Mô hình xây dựng và hiệu chỉnh	6
3.2 Mô hình thác nước	6
3.3 Mô hình bản mẫu	7
3.4 Mô hình tăng trưởng	8
3.5 Mô hình tăng trưởng nhiều rủi ro	9
3.6 Mô hình đồng bộ và ổn định	10
3.7 Mô hình xoắn ốc	10
3.8 Mô hình vòi phun nước	12
3.9 Mô hình mã nguồn mở	13
3.10 Mô hình lặp và tăng trưởng	13
3.11 Mô hình cây tiến hóa	15
3.12 Mô hình phát triển nhanh	15
3.13 Mô hình phát triển ứng dụng nhanh	15
3.14 Mô hình V	16
4. TỔ CHỨC NHÓM LÀM VIỆC	17
4.1 Nhóm làm việc dân chủ	17
4.2 Nhóm làm việc cổ điển	17
4.3 Nhóm làm việc hiện đại	17
4.4 Nhóm làm việc đồng bộ hóa và ổn định	18
4.5 Nhóm làm việc nhanh	19
4.6 Nhóm làm việc mã nguồn mở	19
4.7 Nhóm làm việc P-CMM	19
4.8 Nhóm làm việc SWAT	19
4.9 Nhóm làm việc phân cấp	19
4.10 Nhóm làm việc ma trận	20

4.11 Nhóm làm việc cấu trúc mở.....	20
5. CÔNG CỤ PHẦN MỀM	20
6. KIỂM THỬ	22
6.1 Kiểm thử không dựa trên thực thi.....	22
6.2 Kiểm thử dựa trên thực thi.....	23
7. MÔ ĐUN/ĐỐI TƯỢNG	23
7.1 Độ gắn kết.....	23
7.2 Độ nối kết	24
8. SỬ DỤNG LẠI, DỄ DI CHUYỂN VÀ VẬN HÀNH TƯƠNG TÁC	24
9. ĐÁNH GIÁ PHẦN MỀM.....	25
TÀI LIỆU THAM KHẢO.....	25
CÂU HỎI HƯỚNG DẪN ÔN TẬP	27
ĐỊNH HƯỚNG THẢO LUẬN.....	27
BÀI TẬP THỰC HÀNH.....	28
CHƯƠNG 2. TIẾN TRÌNH PHẦN MỀM	29
1. PHÂN TÍCH YÊU CẦU.....	29
1.1 Phỏng vấn theo cấu trúc.....	29
1.2 Phỏng vấn không theo cấu trúc.....	29
1.3 Gửi bản câu hỏi.....	30
1.4 Khảo sát các biểu bảng	30
1.5 Quay phim	30
1.6 Sử dụng kịch bản	30
1.7 Dựa trên bản mẫu.....	31
2. ĐẶC TẢ/PHÂN TÍCH HỆ THỐNG	32
2.1 Đặc tả không hình thức.....	32
2.2 Đặc tả bán hình thức	32
2.3 Đặc tả hình thức.....	33
3. THIẾT KẾ.....	33
4. CÀI ĐẶT.....	33
5. TÍCH HỢP	34
6. BẢO TRÌ.....	35
TÀI LIỆU THAM KHẢO.....	37

CÂU HỎI HƯỚNG DẪN ÔN TẬP	38
ĐỊNH HƯỚNG THẢO LUẬN.....	38
BÀI TẬP THỰC HÀNH.....	39
CHƯƠNG 3. TIẾN TRÌNH RUP.....	40
1. RUP LÀ GÌ?.....	40
2. CẤU TRÚC ĐỘNG CỦA RUP.....	41
3. CẤU TRÚC TĨNH CỦA RUP.....	42
TÀI LIỆU THAM KHẢO.....	45
CÂU HỎI HƯỚNG DẪN ÔN TẬP	46
ĐỊNH HƯỚNG THẢO LUẬN.....	46
BÀI TẬP THỰC HÀNH.....	46
CHƯƠNG 4. ƯỚC LƯỢNG CHI PHÍ.....	47
1. XÁC ĐỊNH KÍCH THƯỚC PHẦN MỀM.....	47
1.1 LOC	47
1.2 FFP.....	47
1.3 FP.....	48
1.4 OP	52
1.5 EFP	52
1.6 FTP	53
1.7 UCP.....	53
2. ĐÁNH GIÁ MỨC ĐỘ TIN CẬY.....	54
3. ĐÁNH GIÁ MỨC ĐỘ SẴN SÀNG.....	54
4. ĐÁNH GIÁ THỜI GIAN THAY ĐỔI.....	55
5. ĐÁNH GIÁ PHÂN TÍCH YÊU CẦU.....	55
6. ĐÁNH GIÁ THIẾT KẾ KIẾN TRÚC.....	56
7. ĐÁNH GIÁ THIẾT KẾ KIẾN TRÚC HƯỚNG ĐỐI TƯỢNG.....	57
8. ĐÁNH GIÁ THIẾT KẾ CHI TIẾT	58
9. ĐÁNH GIÁ CÀI ĐẶT.....	58
10. ĐÁNH GIÁ KIỂM THỬ	59
11. ĐÁNH GIÁ BẢO TRÌ	60
12. ƯỚC LƯỢNG THỰC NGHIỆM.....	60
13. COCOMO	61

13.1 COCOMO cơ bản	61
13.2 COCOMO trung gian	61
13.3 COCOMO 2.0.....	63
13.4 Sử dụng mô hình COCOMO 2.0 ước lượng trong sử dụng lại	65
13.5 Đếm số lượng điểm chức năng trong COCOMO 2.0.....	66
13.6 Chi tiết các yếu tố về sản phẩm trong COCOMO 2.0.....	69
13.7 Đếm số dòng mã lệnh trong COCOMO 2.0.....	71
13.8 Tổng kết các công thức ước lượng trong COCOMO 2.0	72
13.9 Sử dụng các hệ số nhân	73
13.10 Ước lượng dựa trên phương trình.....	73
13.11 Ước lượng dựa trên các kỹ thuật khác.....	74
TÀI LIỆU THAM KHẢO.....	74
CÂU HỎI HƯỚNG DẪN ÔN TẬP	75
ĐỊNH HƯỚNG THẢO LUẬN.....	75
BÀI TẬP THỰC HÀNH.....	75
CHƯƠNG 5. XÁC ĐỊNH TRỊ GIÁ PHẦN MỀM THEO CÔNG VĂN	
3364/BTTTT	77
1. XÁC ĐỊNH TRỊ GIÁ PHẦN MỀM	77
2. GIÁ TRỊ NỖ LỰC THỰC TẾ.....	77
3. TÍNH ĐIỂM TÁC NHÂN	78
4. TÍNH ĐIỂM TRƯỜNG HỢP SỬ DỤNG	78
5. TÍNH HỆ SỐ PHỨC TẠP KỸ THUẬT – CÔNG NGHỆ	80
6. TÍNH HỆ SỐ PHỨC TẠP MÔI TRƯỜNG	81
7. DỰ KIẾN TRÌNH ĐỘ VÀ KINH NGHIỆM CẦN CÓ CỦA NHÂN CÔNG LAO ĐỘNG.....	81
8. TÍNH HỆ SỐ TÁC ĐỘNG MÔI TRƯỜNG VÀ NHÓM LÀM VIỆC, HỆ SỐ PHỨC TẠP VỀ MÔI TRƯỜNG, XÁC ĐỊNH ĐỘ ỔN ĐỊNH KINH NGHIỆM VÀ NỘI SUY THỜI GIAN LAO ĐỘNG.....	82
9. TÍNH HỆ SỐ TÁC ĐỘNG MÔI TRƯỜNG VÀ NHÓM LÀM VIỆC.....	83
10. TÍNH ĐỘ ỔN ĐỊNH KINH NGHIỆM	84
11. TÍNH THỜI GIAN LAO ĐỘNG.....	85
12. MỨC LƯƠNG LAO ĐỘNG BÌNH QUÂN.....	85
TÀI LIỆU THAM KHẢO.....	85

CÂU HỎI HƯỚNG DẪN ÔN TẬP	86
ĐỊNH HƯỚNG THẢO LUẬN.....	87
BÀI TẬP THỰC HÀNH.....	87
PHỤ LỤC A. CÁC PHẦN TỬ TRONG RUP	88
1. CÁC VAI TRÒ	88
1.1 Các vai trò phân tích (analyst roles/workers).....	88
1.2 Các vai trò phát triển (developer roles)	88
1.3 Các vai trò quản lý (manager roles).....	89
1.4 Các vai trò kiểm thử (tester roles)	90
1.5 Các vai trò sản xuất và hỗ trợ (production and support roles)	90
1.6 Các vai trò khác (additional roles).....	91
2. CÁC TÁC VỤ (artifacts).....	91
PHỤ LỤC B. KẾ HOẠCH VỀ PHẦN MỀM	95
PHỤ LỤC C. TÍNH TRỊ GIÁ PHẦN MỀM “Website trên Internet”	100

DANH SÁCH BẢNG

Bảng 1: So sánh chi phí các dự án phần mềm.....	2
Bảng 2: Trọng số cho các dạng chức năng.....	48
Bảng 3: Các nhân tố ảnh hưởng đến xác định điểm chức năng.....	49
Bảng 4: Xác định độ khó cho các dạng chức năng tập tin.....	50
Bảng 5: Chuyển đổi giữa LOC và FP (tính trung bình).....	50
Bảng 6: Chuyển đổi giữa LOC và FP.....	51
Bảng 7: Trọng số cho dạng đối tượng giao diện.....	52
Bảng 8: Ví dụ về tính kích thước phần mềm cho các trường hợp sử dụng.....	54
Bảng 9: Mức độ khó khi phát triển sản phẩm (COCOMO cơ bản).....	61
Bảng 10: Mức độ khó khi phát triển sản phẩm (COCOMO trung gian).....	61
Bảng 11: Các hệ số nhân của mô hình COCOMO trung gian.....	62
Bảng 12: Các giá trị ước lượng cho COCOMO 2.0.....	63
Bảng 13: Giá trị các hệ số điều chỉnh cho COCOMO 2.0.....	63
Bảng 14: Đếm điểm đối tượng màn hình.....	64
Bảng 15: Đếm điểm đối tượng báo biểu.....	64
Bảng 16: Trọng số độ khó các dạng đối tượng (điểm đối tượng).....	64
Bảng 17: Xác định tỷ suất năng suất.....	64
Bảng 18: Xác định mức độ hiểu biết SU.....	66
Bảng 19: Xác định mức độ đánh giá và đồng bộ hóa AA.....	66
Bảng 20: Trọng số độ khó trong COCOMO 2.0.....	67
Bảng 21: Xác định mức độ cho ILF và EIF.....	67
Bảng 22: Xác định mức độ cho EO và EQ.....	67
Bảng 23: Xác định mức độ cho EI.....	67
Bảng 24: Các hệ số điều chỉnh trong COCOMO 2.0.....	68
Bảng 25: Hệ số điều chỉnh nhân công cho thời kỳ Post-Architecture.....	69
Bảng 26: Hệ số nhân cho các thời kỳ Early Design và Post-Architecture.....	73
Bảng 27: Xác định trị giá phần mềm.....	77
Bảng 28: Xác định điểm tác nhân.....	78
Bảng 29: Xác định trọng số tác nhân.....	78

Bảng 30: Xác định điểm trường hợp sử dụng	79
Bảng 31: Xác định trọng số BMT	79
Bảng 32: Xác định hệ số kỹ thuật công nghệ TFW	80
Bảng 33: Dự kiến trình độ và kinh nghiệm	81
Bảng 34: Xác định hệ số tác động.....	82
Bảng 35: Xác định hệ số tác động môi trường và nhóm làm việc	83
Bảng 36: Xác định giá trị ổn định kinh nghiệm	84
Bảng 37: Xác định giá trị thời gian lao động	85

DANH SÁCH HÌNH

Hình 1: Kết quả thực hiện của hơn 9000 dự án hoàn thành trong năm 2004.	1
Hình 2: Giá thành của các giai đoạn trong chu trình sống của phần mềm.....	2
Hình 3: Chi phí phải trả để điều chỉnh lỗi.	2
Hình 4: So sánh hai phương pháp cài đặt (a) cấu trúc và (b) hướng đối tượng.	3
Hình 5: Tiến trình phần mềm với kiểm soát quản lý.....	4
Hình 6: Các giai đoạn trong chu trình sống của phần mềm.	4
Hình 7: Năm mức của CMM.....	5
Hình 8: Nội dung chi tiết của năm mức trong CMM.	5
Hình 9: Mô hình xây dựng và hiệu chỉnh.....	6
Hình 10: Mô hình thác nước.....	7
Hình 11: Mô hình bản mẫu.....	8
Hình 12: Mô hình tăng trưởng.....	9
Hình 13: Mô hình tăng trưởng nhiều rủi ro.	9
Hình 14: Mô hình xoắn ốc đơn giản.....	11
Hình 15: Mô hình xoắn ốc đơn giản (trình bày lại một phần).	11
Hình 16: Mô hình xoắn ốc đầy đủ.....	12
Hình 17: Mô hình vòi phun nước.	12
Hình 18: Mô hình mã nguồn mở.	13
Hình 19: Mô hình lặp và tăng trưởng.	14
Hình 20: Ba bước lặp của quá trình tăng trưởng B.	14
Hình 21: Mô hình cây tiến hóa.....	15
Hình 22: Mô hình phát triển ứng dụng nhanh.	16
Hình 23: Mô hình V.	16
Hình 24: Cấu trúc nhóm lập trình hiện đại.....	18
Hình 25: Cấu trúc tổ chức quản lý kỹ thuật cho các dự án lớn.....	18
Hình 26: Cấu trúc tổ chức quản lý kỹ thuật cho các dự án lớn với việc phi tập trung hóa các quyết định và các kênh giao tiếp kỹ thuật.	18
Hình 27: Cấu trúc tổ chức quản lý phân cấp.	20
Hình 28: Cấu trúc tổ chức quản lý dạng ma trận.....	20
Hình 29: (a) Công cụ, (b) nhóm công cụ, (c) môi trường.	21

Hình 30: Các dạng phiên bản khác nhau (a) revisions (b) variations (c) mixte..	21
Hình 31: Các thành phần cấu hình của một chương trình thực thi.	22
Hình 32: Sử dụng bản mẫu cho phân tích yêu cầu.....	31
Hình 33: Các khoảng thời gian cho mỗi dạng bảo trì.	36
Hình 34: Tiến trình RUP với hai trục thời gian và nội dung.....	40
Hình 35: Vai trò, hoạt động và tác vụ trong RUP.	42
Hình 36: Gắn kết con người và vai trò.	43
Hình 37: Dòng công việc.....	44
Hình 38: Đếm số dòng mã lệnh.....	71
Hình 39: Các tác vụ mô hình hóa nghiệp vụ.....	91
Hình 40: Các tác vụ yêu cầu.....	92
Hình 41: Các tác vụ phân tích và thiết kế.	92
Hình 42: Các tác vụ cài đặt.	93
Hình 43: Các tác vụ kiểm thử.....	93
Hình 44: Các tác vụ triển khai.....	93
Hình 45: Các tác vụ quản lý cấu hình và thay đổi cấu hình.....	94
Hình 46: Các tác vụ quản lý dự án.	94
Hình 47: Các tác vụ môi trường.	94
Hình 48: ISO 9001: Các hệ thống chất lượng.	95
Hình 49: Kế hoạch đảm bảo chất lượng phần mềm [IEEE Standard 730].	96
Hình 50: Đặc tả yêu cầu phần mềm [IEEE Standard 830].....	97
Hình 51: Kế hoạch thẩm tra và hợp lệ hóa phần mềm [IEEE Standard 1012]. ...	97
Hình 52: Kế hoạch quản lý cấu hình [IEEE Standard 90b].	98
Hình 53: Kế hoạch quản lý dự án phần mềm [IEEE Standard 1058].	99

MỘT SỐ THUẬT NGỮ

Artifacts	Tác vụ
Capability Maturity Models	Mô hình khả trưởng
Code-and-Fix Model	Mô hình xây dựng và hiệu chỉnh
Computer-Aided Software Engineering	Công cụ hỗ trợ phát triển phần mềm
Configuration Control	Kiểm soát cấu hình
Constructive Cost Model (COCOMO)	Mô hình chi phí kiến tạo
Cohesion Level	Mức độ gắn kết
Coupling Level	Mức độ nối kết
Data Dictionary	Từ điển dữ liệu
Egoless Programmer	Lập trình viên bản ngã
Evolution-Tree Model	Mô hình cây tiến hóa
Extreme Programming and Agile Process	Mô hình phát triển nhanh
Files – Flows – Processes	Tập tin – Luồng – Tiến trình
Fountain Model	Mô hình vòi phun nước
Function Point	Điểm chức năng
Incremental Model	Mô hình tăng trưởng
Iterative-and-Incremental Model	Mô hình lặp và tăng trưởng
Kilo or Thousand Delivered Source Instructions (KDSI/KLOC)	Số dòng mã lệnh tính theo đơn vị ngàn
Lines of Code	Dòng mã lệnh
Maintenance Programmer	Lập trình viên bảo trì
Object Point	Điểm đối tượng
Open Source Model	Mô hình mã nguồn mở
Rational Unified Process® (RUP)	Tiến trình hợp nhất Rational
Rapid Application Development	Mô hình phát triển ứng dụng nhanh
Rapid Prototyping Model	Mô hình bản mẫu
Requirements Analysis Team (RAT)	Nhóm phân tích yêu cầu
Risk Incremental Model	Mô hình tăng trưởng nhiều rủi ro

Role/Worker	Vai trò
Self –Documenting Code	Chú thích tự thân
Synchronize-and-Stabilize Model	Mô hình đồng bộ và ổn định
Spiral Model	Mô hình xoắn ốc
Software Development	Phát triển phần mềm
Software Engineering	Công nghệ phần mềm, kỹ thuật phần mềm, kỹ nghệ phần mềm
Software Process	Tiến trình phần mềm
Software Project Management Plan	Kế hoạch quản lý dự án phần mềm
Software Quality Assurance (SQA)	Bảo đảm chất lượng phần mềm
Unified Modeling Language (UML)	Ngôn ngữ mô hình hóa hợp nhất
Use Case Point	Điểm trường hợp sử dụng
Validation	Công nhận hợp lệ
Verification	Thẩm tra
Version	Phiên bản
Waterfall Model	Mô hình thác nước

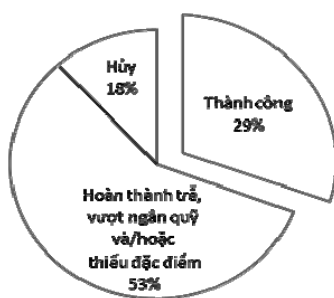
CHƯƠNG 1

TỔNG QUAN

1. PHẠM VI

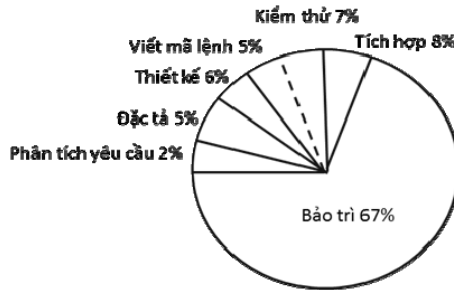
Thuật ngữ công nghệ phần mềm (Software Engineering - SE) được đề xuất đầu tiên vào năm 1967. Từ đó đến nay, sự phát triển của phần mềm đã có những bước phát triển mạnh mẽ. Sự phát triển của phần mềm luôn chịu ảnh hưởng của sự phát triển về phần cứng và hệ điều hành: hệ điều hành đa nhiệm (1960s), bộ nhớ ảo (1970s), đa xử lý (multiprocessor), hệ điều hành phân tán (mạng), hệ điều hành cho di động, hệ điều hành cho hệ thống nhúng, v.v.... Song song đó, vấn đề bảo trì phần mềm luôn đặt ra những thách thức không nhỏ.

Khía cạnh kinh tế trong quá trình phát triển phần mềm luôn là vấn đề quan trọng, nhất là sự lựa chọn kỹ thuật thực hiện nhanh hơn để giảm giá thành, sự ảnh hưởng của kỹ thuật mới lên công ty phần mềm (khó bảo trì, thời gian huấn luyện, kinh nghiệm làm việc trên kỹ thuật mới chưa nhiều), sự phụ thuộc vào lựa chọn của khách hàng, luật bản quyền. Hình 1 cho thấy tỉ lệ thành công một cách trọn vẹn chỉ chiếm khoảng 29% số lượng các dự án phần mềm.



Hình 1: Kết quả thực hiện của hơn 9000 dự án hoàn thành trong năm 2004 [6].

Khía cạnh bảo trì trong chu trình sống của phần mềm (phân tích yêu cầu, phân tích hệ thống/đặc tả, thiết kế, cài đặt, tích hợp, bảo trì, kết thúc hoạt động) ảnh hưởng rất lớn lên giá thành phần mềm. Trên Hình 2 ta thấy giai đoạn bảo trì, về nguyên tắc, chiếm đến 67% thời gian thực hiện phần mềm (xem thêm Bảng 1).

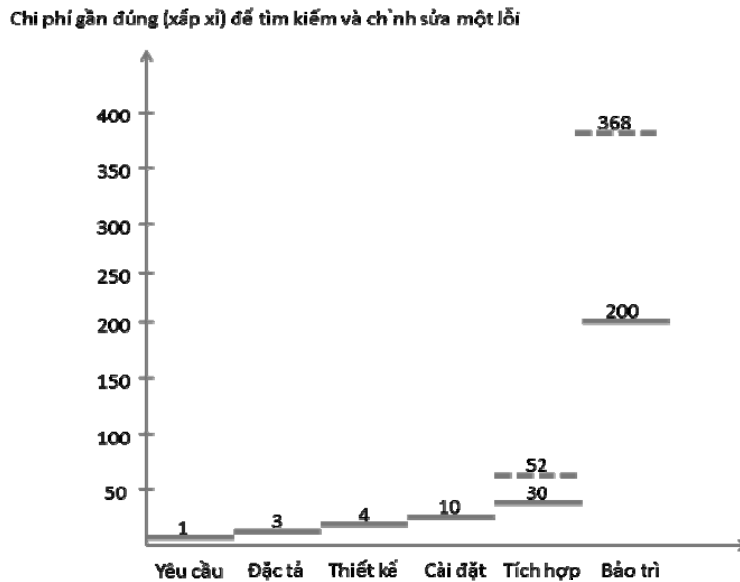


Hình 2: Giá thành của các giai đoạn trong chu trình sống của phần mềm.

Bảng 1: So sánh chi phí các dự án phần mềm.

	Các dự án khác nhau từ 1976 đến 1981	132 dự án gần đây nhất của Hewlett Packard
Giai đoạn yêu cầu và đặc tả	22%	18%
Giai đoạn thiết kế	18%	19%
Giai đoạn cài đặt	36%	34%
Giai đoạn tích hợp	24%	29%
Phát triển	33%	25%
Bảo trì	67%	75%

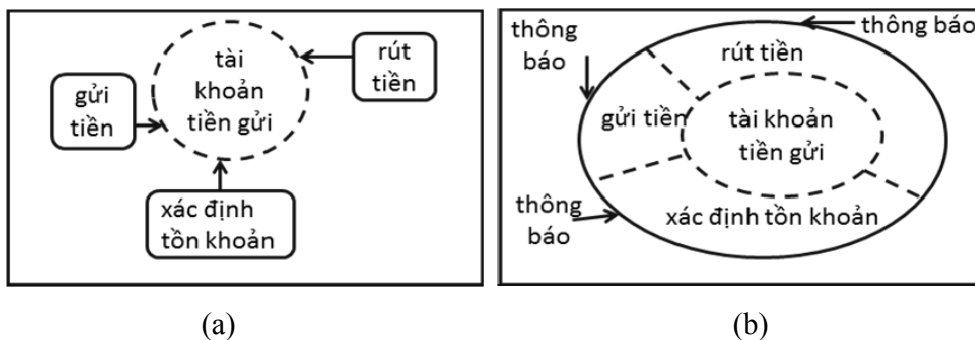
Cần chú ý là sai sót tại các giai đoạn trước sẽ ảnh hưởng rất lớn đến các giai đoạn sau, tạo ra các lỗi. Công việc sửa chữa các lỗi càng sớm càng tốt là rất quan trọng (xem Hình 3).



Hình 3: Chi phí phải trả để điều chỉnh lỗi.

Đội ngũ lập trình là một vấn đề cần được quan tâm sát sao trong quá trình thực hiện phần mềm trên cơ sở hình thành từng nhóm làm việc chuyên biệt trong từng lĩnh vực. Một số vấn đề nảy sinh cần giải quyết như cách chia sẻ các phần công việc, mối quan hệ, sự giao tiếp giữa các thành viên với nhau, kỹ thuật tổ chức và quản lý đội ngũ phát triển phần mềm (lập trình viên, đặc tả viên, thiết kế viên,...), sự ràng buộc lẫn nhau giữa các thành viên (cùng nhóm, khác nhóm,...), cách đánh giá thời gian làm việc, cách đánh giá hiệu quả công việc, cách đánh giá về kinh nghiệm thực hiện công việc.

Một trong những tiếp cận xây dựng phần mềm quan trọng hiện nay là hướng đối tượng xem dữ liệu và tác động có vai trò quan trọng như nhau, hướng kỹ thuật này còn được gọi là thiết kế hướng trách nhiệm hay thiết kế theo hợp đồng. Hình 4 cho thấy sự khác nhau cơ bản giữa hướng cài đặt theo cấu trúc trên cơ sở can thiệp trực tiếp vào dữ liệu và theo hướng đối tượng chỉ can thiệp vào dữ liệu thông qua việc gửi các thông điệp.



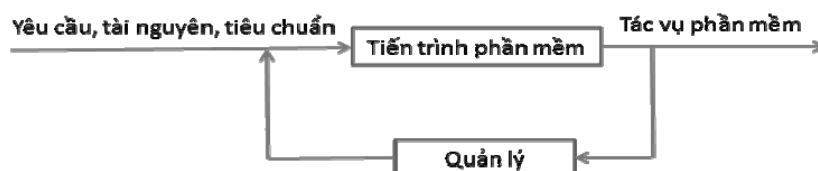
Hình 4: So sánh hai phương pháp cài đặt (a) cấu trúc và (b) hướng đối tượng.

Một số thuật ngữ cần quan tâm như: phần mềm phải bao gồm mã lệnh dưới dạng máy có thể đọc được; các dạng tài liệu đặc tả, thiết kế, luật và sổ sách về chi phí; kế hoạch quản lý dự án phần mềm và các tài liệu quản lý khác; các dạng tài liệu hướng dẫn sử dụng. Bên cạnh đó chương trình là một đoạn mã lệnh có thể tự thực thi được trong khi hệ thống là tập hợp các chương trình liên quan với nhau còn sản phẩm là một mẫu bình thường của phần mềm và là kết quả đạt được sau một tiến trình phát triển phần mềm. Một vài thuật ngữ khác như sản xuất phần mềm bao gồm hai giai đoạn phát triển phần mềm và bảo trì, đồng thời cần phân biệt lỗi (error) hay có lỗi (bug) là hai vấn đề khác nhau.

2. TIẾN TRÌNH PHẦN MỀM

Tiến trình phần mềm (software process) là cách thức tạo ra phần mềm, mỗi công ty có tiến trình phần mềm riêng (xem Hình 5, Hình 6). Khách hàng (client) là cá nhân hay công ty đặt hàng sản phẩm, nhà phát triển (developer) là các thành viên của công ty có trách nhiệm phát triển phần mềm đã được đặt hàng nhằm quản

xuýt toàn bộ các công việc hoặc có trách nhiệm một phần như thiết kế, cài đặt,... Các dạng quan hệ giữa khách hàng và nhà phát triển bao gồm dạng phần mềm nội bộ (internal software) và phần mềm hợp đồng (contract software). Người sử dụng (user) là một hay nhiều cá nhân thay mặt khách hàng để sử dụng sản phẩm. Phát triển phần mềm (software development) bao gồm tất cả các công việc tạo ra sản phẩm trước khi nó được chuyển sang giai đoạn bảo trì.



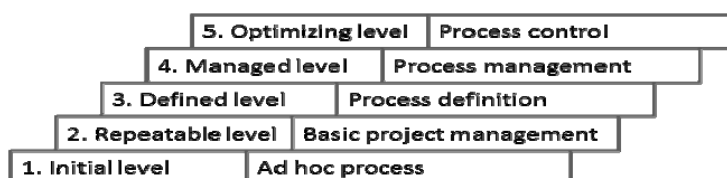
Hình 5: Tiến trình phần mềm với kiểm soát quản lý.



Hình 6: Các giai đoạn trong chu trình sống của phần mềm.

Cần quan tâm đến một số khía cạnh trong sản xuất phần mềm. Độ phức tạp (complexity) là một thuộc tính của phần mềm tác động trên tiến trình phần mềm và cả công tác quản lý tiến trình, có thể biểu diễn bằng toán học và vật lý và ảnh hưởng đến công tác bảo trì. Sự thích ứng (conformity) xác định phần mềm phải thích ứng được với các thiết bị sẵn có (không phải các thiết bị đáp ứng phần mềm), thích ứng tốt với phần cứng phục vụ phần mềm. Dễ chuyển đổi (changeability) xác định phần mềm phải thay đổi theo thực tiễn, mở rộng các chức năng ban đầu với thực tế là thay đổi phần mềm dễ hơn thay đổi về phần cứng và sau đó phần cứng thay đổi theo sự phát triển của phần mềm hoặc công

nghệ. Tính vô hình (invisibility) nhằm giấu các công đoạn phức tạp khi thực hiện phần mềm, dễ dàng thuyết minh, thuyết phục khách hàng, dễ dàng giao tiếp giữa các bộ phận thực hiện phần mềm, sử dụng các phương pháp, công cụ trực quan sinh động. Việc nhanh chóng tạo phần mềm mới từ các bộ phận hay công cụ có sẵn (silver bullet) là nhằm nhanh chóng chuyển đổi chức năng của sản phẩm, giảm thời gian cũng như chi phí thực hiện phần mềm, sử dụng mô hình chuyển đổi nhanh. Cần chú ý thuật ngữ lỗi trên 1000 dòng mã lệnh tương đương hợp ngữ (faults per million equivalent assembler source – MEASL).



Hình 7: Năm mức của CMM.



Hình 8: Nội dung chi tiết của năm mức trong CMM.

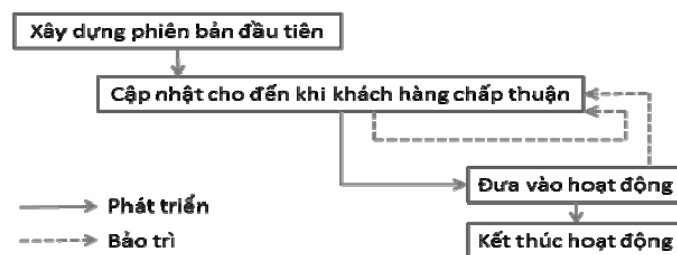
Cải tiến tiến trình phần mềm với mô hình khả trưởng CMM (Capability Maturity Models) là nhóm các chiến lược liên quan với nhau nhằm cải tiến tiến trình phần mềm. Có các nội dung cho phần mềm SW-CMM (Software), cho quản lý nguồn nhân lực P-CMM (People), cho công nghệ hệ thống SE-CMM (System Engineering), cho phát triển sản phẩm tích hợp IPD-CMM (Integrated Product Development) và cho đạt được sản phẩm SA-CMM (Software Aquisition).

CMMs là tiêu chuẩn khởi điểm cho các tiêu chuẩn quốc tế về sau như ISO 9000 (International Standards Organization 9000-series standards) gồm 5 chuẩn áp dụng rộng rãi cho các hoạt động công nghiệp: thiết kế (design), phát triển (development), sản xuất (production), cài đặt (installation) và bảo dưỡng (servicing). ISO 9001 dành cho chất lượng sản phẩm [ISO 9001, 1987], áp dụng ISO 9001 cho phần mềm ISO 9000-3 [ISO 9000-3, 1991]; SPICE (Software Process Improvement Capability dEtermination) tương tự như SW-CMM và ISO 9000 thống nhất hai chuẩn từ 06/1997: ISO/IEC 15504 hay 15504.

3. CÁC MÔ HÌNH CHU TRÌNH SỐNG CỦA PHẦN MỀM

3.1 Mô hình xây dựng và hiệu chỉnh

Mô hình xây dựng và hiệu chỉnh (code-and-fix model) không có đặc tả hay thiết kế, chỉ đơn giản là làm đi làm lại cho đến khi nào đáp ứng được yêu cầu của khách hàng. Thường sử dụng trong các bài tập lập trình từ 100 đến 200 dòng mã lệnh.

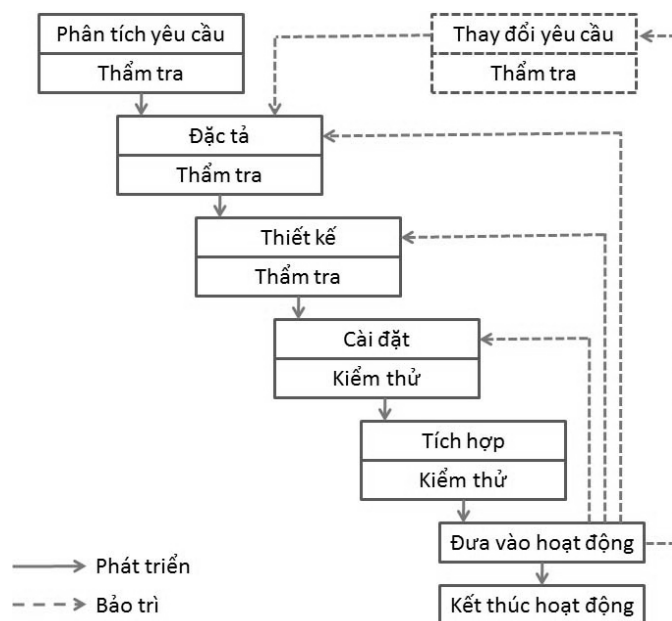


Hình 9: Mô hình xây dựng và hiệu chỉnh.

3.2 Mô hình thác nước

Mô hình thác nước (waterfall model) cho phép các lỗi ở một số giai đoạn trước được phản hồi bởi các giai đoạn sau và mỗi giai đoạn chỉ được xem là hoàn thành sau khi đã có đầy đủ tài liệu cho giai đoạn đó và được nhóm đảm bảo chất lượng phần mềm (SQA) chấp thuận. Các bước tiến hành bắt đầu với các yêu cầu được xác định và kiểm chứng bởi khách hàng và nhóm SQA, các đặc tả được kiểm chứng bởi nhóm SQA và gửi cho khách hàng, khung kế hoạch quản lý dự án phần mềm (SPMP) và bảng thời gian làm việc chi tiết được lập. Giai đoạn thiết kế bắt đầu sau khi khách hàng đồng ý về giá thành và thời gian thực hiện, thực

hiện cài đặt và tích hợp, khách hàng cho hoạt động thử; chấp nhận sản phẩm, chuyển sang giai đoạn bảo trì. Mô hình này có ưu điểm là tính kỷ luật cao, quy định tốt về tài liệu cho mỗi giai đoạn, kiểm chứng cẩn thận bởi nhóm SQA, được ứng dụng rộng rãi. Nhược điểm của mô hình là quá nhiều kiểm thử, thẩm tra và tài liệu nên khó hình dung và khó hiểu đối với khách hàng.



Hình 10: Mô hình thác nước.

3.3 Mô hình bản mẫu

Mô hình bản mẫu (*rapid prototyping model*) là mô hình hoạt động có chức năng tương đương với một tập hợp con của sản phẩm, được sử dụng khi cần thực hiện một cách nhanh chóng một sản phẩm phần mềm trên cơ sở phần mềm tương tự hay phần mềm mà nhóm thực hiện đã có nhiều kinh nghiệm. Chẳng hạn như nếu chức năng sản phẩm là trả tiền tài khoản, nhận tiền từ tài khoản và xếp hàng vào kho thì việc tạo nhanh bản mẫu có thể bao gồm các công việc của sản phẩm như: màn hình nhập liệu, in các báo cáo nhưng không có các công việc như cập nhật tập tin hay bắt các lỗi xuất hiện.

Đầu tiên là xây dựng bản mẫu của mô hình, tạo điều kiện cho khách hàng và người sử dụng tương lai tương tác với mô hình và thử nghiệm, chuyển sang giai đoạn đặc tả sau khi khách hàng đã chấp thuận rằng các yêu cầu cần thiết đã có trong quá trình xây dựng nhanh bản mẫu. Yêu cầu của mô hình là thực hiện càng nhanh càng tốt để tăng tốc độ của tiến trình phát triển phần mềm.